UNCLASSIFIED MASTER COPY - FOR REPRODUCTION PURPOSES 8

SECURITY CLASSIFICATION OF THIS PAGE

# REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release;
distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)
ARO 24369.39-MA-SDI

6a. NAME OF PERFORMING ORGANIZATION
Purdue University

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
U. S. Army Research Office

6c. ADDRESS *(City, State, and ZIP Code)*
Lafayette, IN 47907

7b. ADDRESS *(City, State, and ZIP Code)*
P. O. Box 12211
Research Triangle Park, NC 27709-2211

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
U. S. Army Research Office

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
DAAL03-86-K-0106

8c. ADDRESS *(City, State, and ZIP Code)*
P. O. Box 12211
Research Triangle Park, NC 27709-2211

10. SOURCE OF FUNDING NUMBERS

| PROGRAM ELEMENT NO. | PROJECT NO. | TASK NO. | WORK UNIT ACCESSION NO. |
|---|---|---|---|
| | | | |

11. TITLE *(Include Security Classification)*
Advanced Parallel Systems

12. PERSONAL AUTHOR(S)
John Rice

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM 8/1/86 TO 9/30/89

14. DATE OF REPORT *(Year, Month, Day)*
November 1989

15. PAGE COUNT
7

16. SUPPLEMENTARY NOTATION
The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

17. COSATI CODES

| FIELD | GROUP | SUB-GROUP |
|---|---|---|
| | | |
| | | |

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Computing Power, Parallel Systems, Algorithms, Advanced Parallel Systems, Architectures

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The objective of this work was to explore algorithms and their implementation for future advanced parallel systems. These systems are assumed to have hundreds or even thousands of processors and to be able to concentrate their computing power on one or a small number of tasks. The three principal questions to be explored were:

1) Are there algorithms for the crucial applications which have enough parallelism to allow the power of the advanced parallel systems to be fully exploited?

2) What languages and implementation tools are needed for efficient programming of these algorithms?

3) What are the relative performances of different algorithm types? Of different architecture types? Of different implementation languages?

(OVER)

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL

22b. TELEPHONE *(Include Area Code)*

22c. OFFICE SYMBOL

83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

The research results obtained appear in this report.

(KR)

ARO 24319.39-MA-SDI

# FINAL TECHNICAL REPORT

**ARO Grant DAAL03-86-K-0106**
**Research Sponsored by SDIO/IST and managed by ARO**

## ADVANCED PARALLEL SYSTEMS

**John R. Rice**
**October 16, 1989**

This report covers the activities of John R. Rice (Co-PI) and associates at Purdue University from July 1986 through September 30, 1989. The work done at Yale University under the direction of Co-PI Martin Schultz is not covered here. The activities include (1) 12 papers published in or submitted to technical journals, (2) 2 book chapters, (3) 25 conference presentations with papers in the conference proceedings, (4) 9 technical reports, and two Ph.D. theses.

The objective of this work was to explore algorithms and their implementation for future advanced parallel systems. These systems are assumed to have hundreds or even thousands of processors and to be able to concentrate their computing power on one or a small number of tasks. The three principal questions to be explored were:

1. Are there algorithms for the crucial applications which have enough parallelism to allow the power of the advanced parallel systems to be fully exploited?
2. What languages and implementation tools are needed for efficient programming of these algorithms?
3. What are the relative performances of different algorithm types? Of different architecture types? Of different implementation languages?

The research results obtained are grouped within four areas, basically those described in the original proposal. We state the principal problem for each area and then list the papers, conference presentations, theses and technical reports for each area, followed by a short summary of principal or typical results obtained.

## A. ANALYSIS OF THE PERFORMANCE OF FUTURE COMPUTATIONS

*Principal Problem*: Analyze the practicality of using massive parallelism efficiently in large scale scientific and engineering computations.

1. D.C. Marinescu and J.R. Rice, Domain Oriented Analysis of PDE Splitting Algorithms, J. Info. Sci, 42 (1987), 3–24.
2. D.C. Marinescu and J.R. Rice, Analysis and Modeling of Schwartz Splitting Algorithms for Elliptic PDEs, in *Advances in Computer Methods for Partial Differential Equations*, VI, IMACS (1987), 1–6. Also conference presentation.

3. **D.C. Marinescu and J.R. Rice,** Nonhomogeneous Parallel Computations: Synchronization Analysis of Parallel Algorithms, CSD-TR-683, Purdue University, (1987), 25 pages.
4. **D.C. Marinescu and J.R. Rice,** Modeling Hardware-Software Interaction in Parallel and Distributed Systems Using Stochastic High Level Petri Nets, *IEEE Distr. Proc. News*, 10 (1988), 28–34.
5. **D.C. Marinescu and J.R. Rice,** Synchronization of Non Homogeneous Parallel Computations, in *Parallel Processing for Scientific Computing* (G. Rodrique, ed.), SIAM (1989), 362–367. Also conference presentation.
6. **D.C. Marinescu and J.R. Rice,** On the Effects of Synchronization in Parallel Computing, CSD-TR-750, Purdue University, (1988), 20 pages. Submitted for journal publication.
7. **C. Lin and D.C. Marinescu,** Stochastic High Level Petri Nets and Applications, *IEEE Trans. Computers*, Vol. 37, No. 7 (1988), 815–825.
8. **C. Lin and D.C. Marinescu,** On Stochastic High Level Petri Nets, in *Petri Nets and Performance Models*, IEEE TH0185-9 (1987), 34–43. Also conference presentation.
9. **D.C. Marinescu and J.R. Rice,** A two level asynchronous algorithm for PDEs, in *Aspects of Asynchronous Numerical Computing* (M. Wright, ed.), Elsevier, New York (1989), 22–33.
10. **D.C. Marinescu and J.R. Rice,** Non-Algorithmic Load Imbalance Effects for Domain Decomposition methods on a Hypercube, CSD-TR-832, Purdue University (1988), 24 pages.
11. **D.C. Marinescu and J.R. Rice,** Multilevel asynchronous iterations for PDEs, in *Iterative Methods* (D. Kincaid, ed.) Academic Press (1989), in press.
12. **C. Lin and D.C. Marinescu,** Reachability Trees for High Level Petri Nets With Marking Variables, CSD-TR-857, Purdue University (1989), 13 pages.
13. **C. Lin and D.C. Marinescu,** An Algorithm for Computing S-Invariants for High Level Petri Nets, CSD-TR-860, Purdue University (1989), 14 pages.
14. **C. Lin and D.C. Marinescu,** Stochastic High Level Petri Nets, Reachability Trees and Invariants, *Int'l Journal Microelectronics and Reliability* (1990), to appear.

A novel aspect of performance analysis in this area is that systems and applications can no longer be analyzed in isolation. Complex models describing interactions among massively parallel systems and large parallel applications have to be studied [4].

In our research we have investigated methodologies for analysis of such systems. We have introduced a class of models based upon Stochastic High Level Petri nets, [7], [8], [12], [13], [14] which support a new approach for the analysis of very complex models. Using symmetry relations reflecting the homogeneity of the model, the state space of the model is drastically reduced and analysis of complex systems, unfeasible by other means, become possible. We have applied SHLPN modeling techniques to Schwartz splitting algorithms on architecture with multiple levels of memory [1], [2].

Another direction of our research was directed towards the study of blocking phenomena caused by the algorithmic need to synchronize different threads of control during the parallel execution of an application [3], [5], [9], [10], [11]. An important conclusion we have obtained is that blocking may be the most significant cause of inefficiency in massively parallel systems. To investigate an upper bound for the processor utilization due to load imbalance effects cause by blocking, we have studied the SPMD (Same Program Multiple Data) model of execution suitable to model domain decomposition numerical methods. In this case we have a perfect algorithmic load balance and blocking occurs due to non algorithmic effects like data dependent MFLOP rates, errors and retrys. We have developed a unified model of execution which takes into account blocking, communication and control [6], [10].

## B. BENCHMARKING EXISTING COMPUTATIONS

*Principal Problem*: Analyze the performance of existing parallel software and machines. Develop methodology for benchmarking the performance of scientific and engineering software.

15. **C.E. Houstis, E.N. Houstis, J. Rice and M. Samartzis**, Benchmarking of Bus Multiprocessor Hardware in Large Scale Scientific Computing, in *Advances in Computer Methods for Partial Differential Equations*, VI IMACS (1987), 136–141. Also conference presentation.
16. **E.N. Houstis, C.C. Christara, J.R. Rice and E.A. Vavalis**, Performance of Scientific Software, Chapter 6 in *Mathematical Aspects of Scientific Software* (Rice, ed.), IMA Volume 14, Springer-Verlag (1988), 123–156. Also conference presentation.
17. **C.J. Ribbens and J.R. Rice**, Realistic PDE Solutions for Non-Rectangular Domains, CSD-TR-639, Purdue University (1986), 35 pages.
18. **H.S. McFaddin and J.R. Rice**, Parallel and Vector Problems on the FLEX/32, CSD-TR-661, Purdue University (1987), 85 pages.
19. **E.N. Houstis, J.R. Rice and E.A. Vavalis**, Benchmarking of MIMD Hardware on Subdomain Splitting Elliptic PDE Solvers, CSD-TR-874, Purdue University (1989), 14 pages.
20. **D.C. Marinescu et. al.**, CAPS - A Coding Aid used with the PASM Parallel Processing Systems, *Proc. of the Workshop on Experiences with Building Distributed and Multiprocessor Systems*, IEEE Computer Society Press (1989), to appear.

This work is focused on how current parallel machines actually perform on scientific computations. Specific applications run on specific machines are reported in [15], [16], [18], and [19]. Generally, we see that good parallel efficiency is achieved on a variety of applications. We have also developed tools and analytic methods for benchmarking and performance evaluations [17], [20]. These results strongly suggest that highly efficient parallel execution is feasible for a broad range of applications.

## C. CONTROL OF PARALLEL COMPUTATIONS

*Principal Problem*: Determine how to break computations into nearly equally sized pieces to distribute to a collection of processors. Determine how parallel processors can synchronize and organize their work so as to avoid or minimize bottlenecks.

21. **C.E. Houstis, E.N. Houstis and J.R. Rice**, Partitioning PDE Computations: Methods and Performance Evaluations, *J. Parallel Comp.*, 4 (1987), 143–163. Also conference presentation.
22. **John R. Rice**, Parallelism in Solving PDEs, *Proc. Fall Joint Computer Conf.*, IEEE (1986), 540–546. Also conference presentation.
23. **H.S. McFaddin, C.E. Houstis and E.N. Houstis**, The mapping of parallel multigrid algorithms onto parallel architectures, CSD-TR 699, Purdue University, July 1987.
24. **Calvin J. Ribbens**, A Priori Grid Adaption Strategies for Elliptic PDEs, in *Advances in Computer Methods for Partial Differential Equations* VI, (R. Vichnevetsky and R.S. Stepleman, eds.), IMACS, (1987), 102–107. Also conference presentation.
25. **Greg N. Frederickson**, Distributed Algorithms for Selection in Sets, *J. Computer Syst. Sci.*, (1988), 337–348.
26. **C.C. Christara, A. Hadjidimos, E.N. Houstis, E.A. Vavalis and J.R. Rice**, Line cubic spline collocation methods for elliptic partial differential equations in multidimensions, in *Computational Methods in Flow Analysis*, Vol. 1, (H. Niki and M. Kawahara, eds.), Okayama University Press, Okayama, Japan (1988), 175–182. Also conference presentation.
27. **D.L. Alexandrakis, C.E. Houstis, E.N. Houstis, J.R. Rice and S.M. Samartzis**, The Algorithm Mapper: A System for Modeling and Evaluating Parallel Application/Architecture Pairs, in *Fourth Generation Mathematical Software Systems* (Houstis, Rice and Vichnevetsky, eds.), North Holland, (1989). Also conference presentation.
28. **N. Chrisochoides, C.E. Houstis, E.N. Houstis, S.K. Kortesis, and J.R. Rice**, Automatic Load Balanced Partitioning Strategies for PDE Computations, in *Third International Conference on Supercomputing*, ACM Press (1989). Also conference presentation.
29. **D.C. Marinescu and W. Szpankowski**, A Safe State Approach in Real-Time Systems Scheduling, *Sixth IEEE Workshop on Real-Time Operating Systems*, Carnegie Mellon University, IEEE Computer Society Press (1989), 54–60.
30. **D.C. Marinescu, J. Lumpp, T.L. Casavant, and H.J. Siegel**, A Model for Monitoring and Debugging Parallel and Distributed Software, *Proc. Computer Software and Applications Conference*, IEEE Computer Society Press (1989), 81–88.
31. **R. Stansifer and D.C. Marinescu**, A Formalism for Critical Path-Analysis of Real-Time Ada Programs, *32nd Symp. on Circuits and Systems*, Univ. of Illinois, Urbana, IL, IEEE Computer Society Press (1989), to appear.
32. **R. Stansifer and D.C. Marinescu**, Petri Net Models of Concurrent Ada Programs, *Proc. of Hawaii Int'l Conf. of System Sciences*, IEEE Computer Society Press (1990), to appear.

A key problem is to decide how to partition applications into separate, parallel tasks and then how to control these tasks for efficient parallel executions. We approach this problem from the very general level [22], [25], [29], [32] to very specific algorithms and techniques [21], [23], [27], [28] to accomplish the partitioning. We also have worked on model execution control and debugging for parallel computations [30], [31]. This work leads to practical, operational systems to partition many scientific applications and provides guidance for an even broader class of applications.

## D. PARALLEL ALGORITHMS FOR PHYSICAL PROBLEMS

*Principal Problem*: Create algorithms that are easily broken into parallel subcomputations and whose total work is near the minimum possible.

33. **John R. Rice**, Parallel Methods for Partial Differential Equations, Chapter 8 in *The Characteristics of Parallel Computation*, (Jamieson, Gannon and Douglass, eds) MIT Press (1987), 209-231.
34. **E.N. Houstis, E.A. Vavalis and J.R. Rice**, Parallelization of a New Class of Cubic Spline Collocation Methods, in *Advances in Computer Methods for Partial Differential Equations*, VI, IMACS (1987), 167-174. Also conference presentation.
35. **Calvin J. Ribbens**, A Fast Grid Adaption Scheme for Elliptic PDEs, in *ACM Trans. Math. Softw.* (1989), to appear.
36. **John R. Rice**, Supercomputing About Physical Objects, in *Supercomputing*, (Houstis, Papatheodorou and Polychronopolos, eds), Lecture Notes in Computer Science 297, Springer-Verlag (1988), 443-455. Also conference presentation.
37. **E.N. Houstis, C.C. Christara and J.R. Rice**, Quadratic Spline Collocation Methods for Two Point Boundary Value Problems, *Intl. J. Numer. Meth. Engin.* (1988) to appear.
38. **C.C. Christara, E.N. Houstis and J.R. Rice**, A Parallel Spline Collocation-Capacitance Method for Elliptic PDEs, in *1988 Int'l Conf. Supercomputing*, ACM Press, New York (1988), 375-385.
39. **E.N. Houstis, J.R. Rice and E.A. Vavalis**, A Schwartz Splitting Variant of Cubic Spline Collocation Methods for Elliptic PDEs, in *Third Conf. Hypercube Concurrent Computers and Appl.*, ACM Press (1988), 1746-1754. Also conference presentation.
40. **Calvin J. Ribbens**, *Domain Mappings: A Tool for the Development of Vector Algorithms for Numerical Solutions of Partial Differential Equations*, Ph.D. thesis, Purdue University, (1987).
41. **Calvin J. Ribbens**, Parallelization of Adaptive Grid Domain Mappings, in *Parallel Processing for Scientific Computing* (G. Rodrique, ed.), SIAM (1989), 196-200. Also conference presentation.
42. **E.N. Houstis and J.R. Rice**, Parallel ELLPACK: An Expert System for the Parallel Processing of Partial Differential Equations, *Math Comp. Simulation*, **31**, (1989), 497–507.

43. **A. Hadjidimos, E.N. Houstis, J.R. Rice, M. Samartzis, E.A. Vavalis,** Semi Iterative Methods on Distributed Memory Multiprocessor Architectures, in *Third Int'l Conf. Supercomputing,* ACM Press (1989). Also conference presentation.

44. **M. Mu and J.R. Rice,** A Grid Based Subtree-Subcube Assignment Strategy for Solving PDEs on Hypercubes, CSD-TR-869, Purdue University (1989), 13 pages. Submitted to a journal.

45. **M. Mu and J.R. Rice,** Solving linear systems with sparse matrices on hypercubes, in *Fourth Conference on Hypercube Concurrent Computers and Applications* (G. Fox, ed.), ACM Press, New York (1989), in press.

46. **J.R. Rice,** Composition of Libraries, Software Parts and Problem Solving Environments, in *Scientific Software* (Cai, Fosdick, Huang, eds.), Tsinghua Univ. Press (1989), 191–203.

47. **C.C. Christara,** *Spline Collocation Methods, Software and Architectures for Linear Elliptic Boundary Value Problems,* Ph.D. Thesis, Purdue University (1988).

48. **J.R. Rice,** Collaborating Modules for Solving PDEs, Conf. on Modeling and Simulation, IMACS (1989), submitted to a journal. Also conference presentation.

49. **M. Mu and J.R. Rice,** Row oriented Gauss elimination on distributed memory multiprocessors, *submitted for publication.*

50. **E.N. Houstis, J.R. Rice, N.P. Chrisochoides, H.C. Karathanasis, P.N. Papachiou, M.K. Samartzis, E.A. Vavalis and K.Wang,** Parallel (//) ELLPACK PDE Solving System, Computer Science Dept. CSD-TR-912, Purdue University, (1989), 60 pages.

Some of these papers develop general principles for highly parallel methods applicable to physical problems. Rice [33] argues that the natural parallelism is the physical world can be used to develop massively parallel computational methods. More recently he describes in [46] and [48] an approach to massive parallelism which also holds the potential to dramatically reduce the cost of software development for the analysis of highly complex physical systems.

Most of the work develops specific parallel algorithms of various types for various physical problems. These include (a) iteration methods [34], [42], [43], (b) direct methods, [44], [45], [49], (c) domain mapping methods [35], [40], [41], and (d) discretizations especially suitable for parallel methods [37], [38], [39], [47]. This work shows that a wide variety of effective parallel methods can be developed and our performance studies show that very good use can be made of parallel computers of various architectures.

Finally, Houstis and Rice [38], [42], [50] have started laying the foundation for automating (via expert systems) much of the complexity involved in using parallel methods on parallel machines.

## E. PERSONNEL

The principal personnel were

John R. Rice (PI) Professor of Computer Science

| | |
|---|---|
| Elias N. Houstis | Professor of Computer Science |
| Dan C. Marinescu | Associate Professor of Computer Science |

They were assisted by the faculty, post-docs and graduate students listed below. Some of these were supported by teaching or fellowships as well as by this project.

| | |
|---|---|
| M. Aboelaze | Visiting Assistant Professor of Computer Sciences |
| N. Chrisochoides | M.S. candidate |
| C.C. Christara | Ph.D. graduate |
| Greg Frederickson | Professor of Computer Science |
| A. Hadjidimos | Visiting Professor of Computer Science |
| C.E. Houstis | Associate Professor of Electrical Engineering |
| H.C. Karathanasis | Visiting Scholar |
| S.K. Kortesis | Visiting Scholar |
| H.S. McFaddin | Ph.D. candidate |
| Mu Mo | Post-doc |
| P. Papachiou | M.S. graduate |
| Calvin J. Ribbens | Ph.D. graduate |
| M. Samartzis | M.S. graduate |
| E.A. Vavalis | Post-doc, Visiting Assistant Professor of Computer Sciences |